SAE AS-4 (JAUS)

SAE AS-4 Technical Committee and the JAUS Working Group

Introduction to and Summary of the

SAE AS-4 JAUS Standard

2008 Defense Standardization Conference
March 6, 2008

Ralph "Woody" English
woodyenglish@devivoast.com

SAE AS-4 (JAUS)

Introduction

- **Scope**

- **Background**

- **Organization**

- **Document Set & Standards**

- **Future Directions**

Statutory Goal

- **Public Law 106-398; 114 Stat.1654A-38...**

- **Goal of one third of ground combat vehicles will be unmanned by 2015**

SAE AS-4 (JAUS)

- **Experiences in theater have proven that robotics can satisfy critical operational needs**

- **GWOT has proven we have only just begun to understand how to leverage unmanned systems in the joint battle space...there will be more for robots to do in future warfare**

- **Robots allow war fighter**

1) **INTEGRATION**

   Projects experience reduction in length of integration from weeks and months to a just few days or, in some cases, hours

2) **SOFTWARE RE-USE**

   JAUS enables companies/vendors to massively reuse software from one project to the next

3) **INTEROPERABILITY**

   Experimentation has demonstrated how JAUS facilitates interoperability between different robots, their OCUs, and payloads

4) **ENHANCED COMPETITION & TECHNOLOGY INSERTION**

   Companies report a leveling of the playing field allowing them to better compete/partner & innovations to bubble up

SAE AS-4 (JAUS)

- **36 Vehicles in Semi-Finals, 5 used JAUS**
- **3rd Place team = Victor-Tango [JAUS]**
- **Take-Aways**
  - **Route & Mission Data Formats**
  - **Protocol (Emergency Remote)**
  - **Processes (safety, operation, …)**
  - **Testing / Qualification / Certification**
  - **Validation of JAUS Communications**

Urban Challenge

- **DoD policy addresses standardization and interoperability as key requirements in its acquisition process**
- **AT&L has supported JAUS since chartering the JAUS effort in 1998**
- **The unmanned ground systems Portfolio (plus UAS/UUV) is growing, making interoperability & technology transfer even more imp**

JAUS is a critical component of the Joint Ground Robotics Portfolio

SAE AS-4 (JAUS)

- **Purpose**: The primary purpose of JAUS is interoperability, e.g., the ability to operate unlike systems with unlike controllers.
- **Product**: A set of standards, specifications and recommendations to facilitate the interoperability of unmanned systems for Government and Industry
- **Payoff**:
  – More efficient development
 – Reduced ownership cost
  – An expanded range of vendors

Exploit existing and future technologies while simultaneously supporting systems evolution to autonomy

- The goal of JAUS is *interoperability* with an emphasis on the *logical communications* between heterogeneous computing systems used for unmanned systems command and control.

- JAUS is a common language enabling internal and external communication between unmanned systems. It incorporates a *component-based*, *message-passing* architecture specifying data formats & protocols that promote stability of *capabilities* by projecting anticipated requirements as well as those currently needed.

- JAUS is a *scalable* language meant to the

A Common Interface Language for Unmanned Systems

9

- **Platform Independence**
  - Supports Interoperability on any platform

- **Mission Isolation**
  - Supports configurable payloads

- **Hardware Independence**
  - Not based on dated technology

- **Technology Independence**
  - Supports technology insertion

- **Operation Independence**
  - Allows the user to determine the operation

- **Communications Independence**
  - No requirement for specific data link

JAUS Remains Flexible, Independent, and Relevant
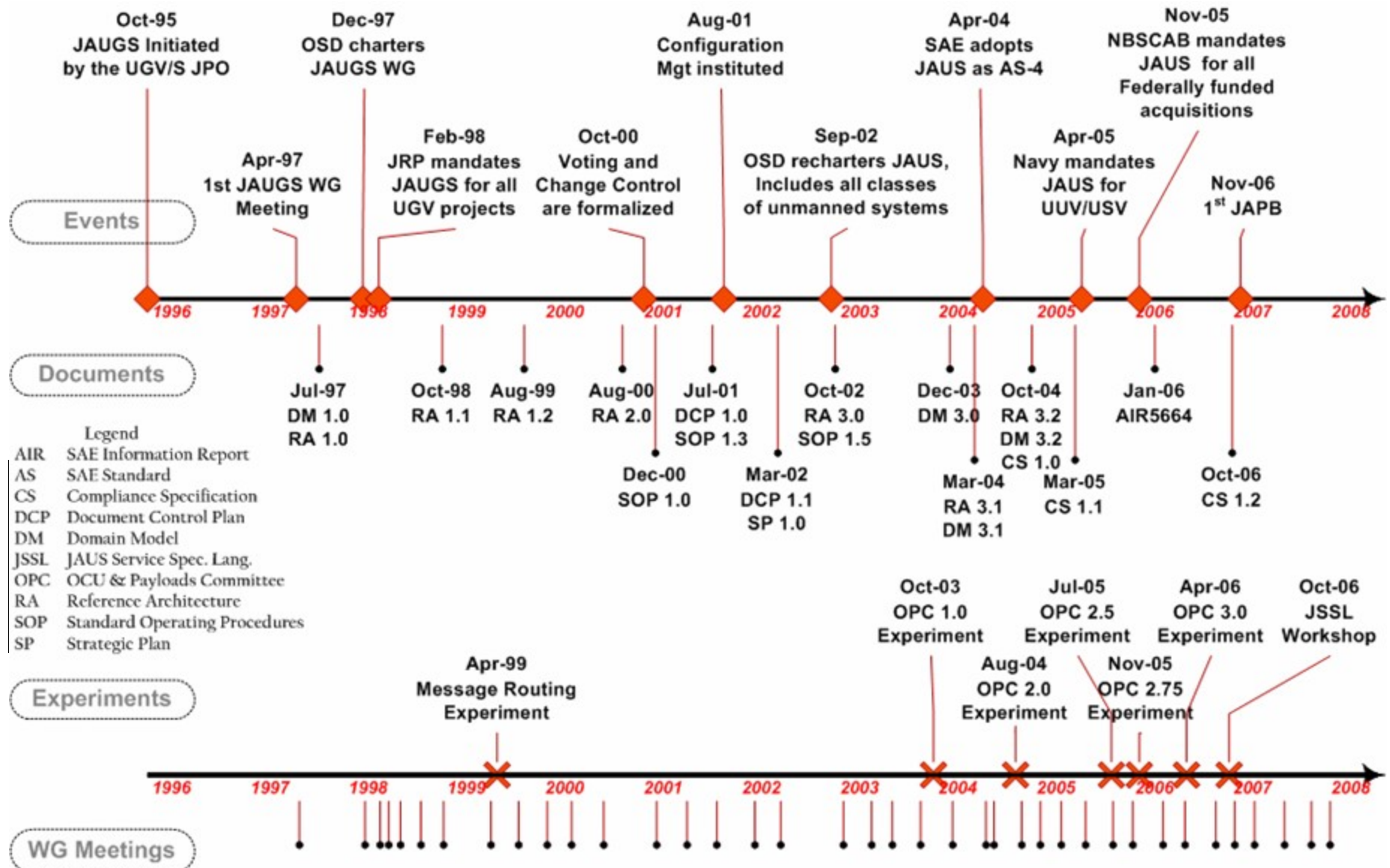
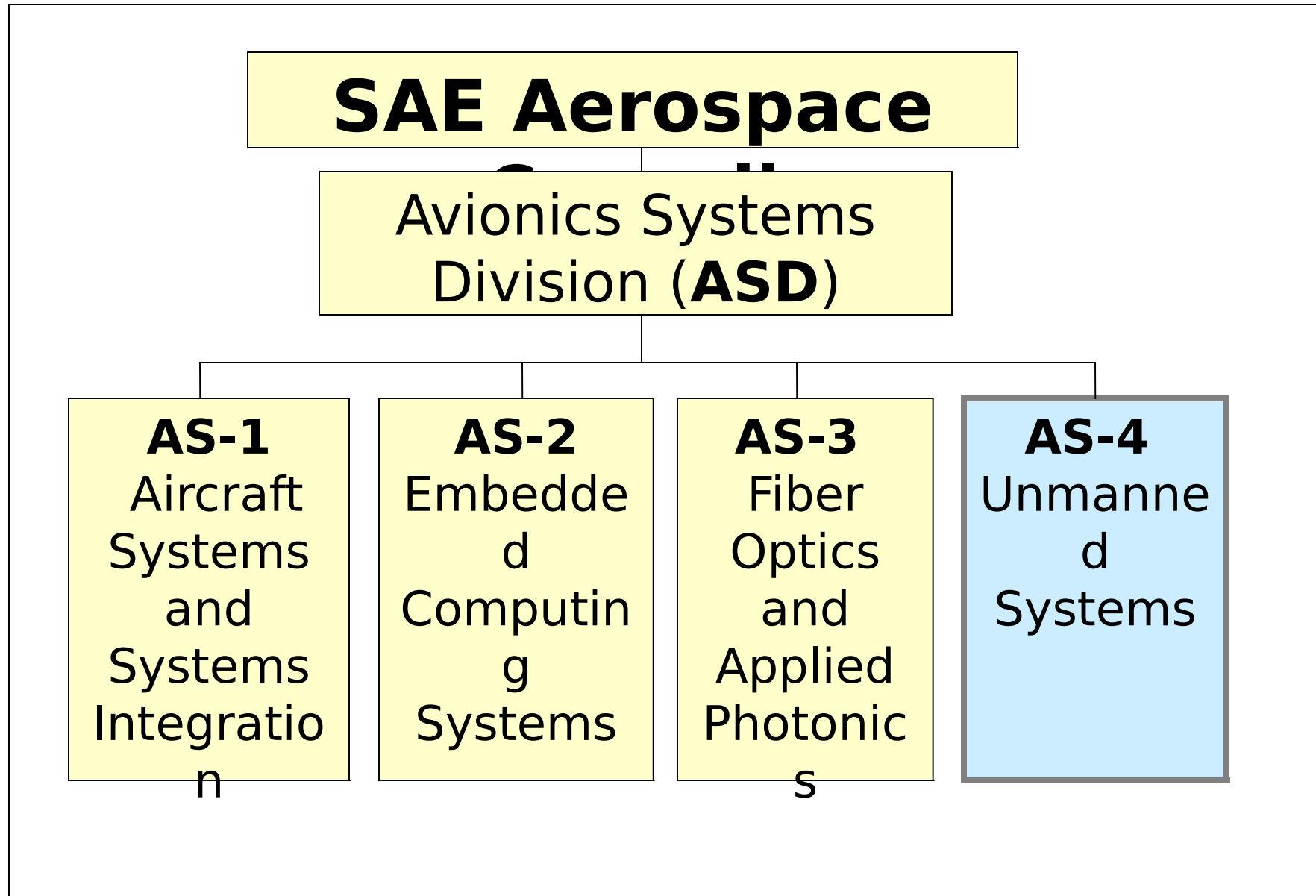**JAUS has over 10 Years of History**

June 2005 Association for Unmanned Vehicle Systems International (AUVSI) Conference
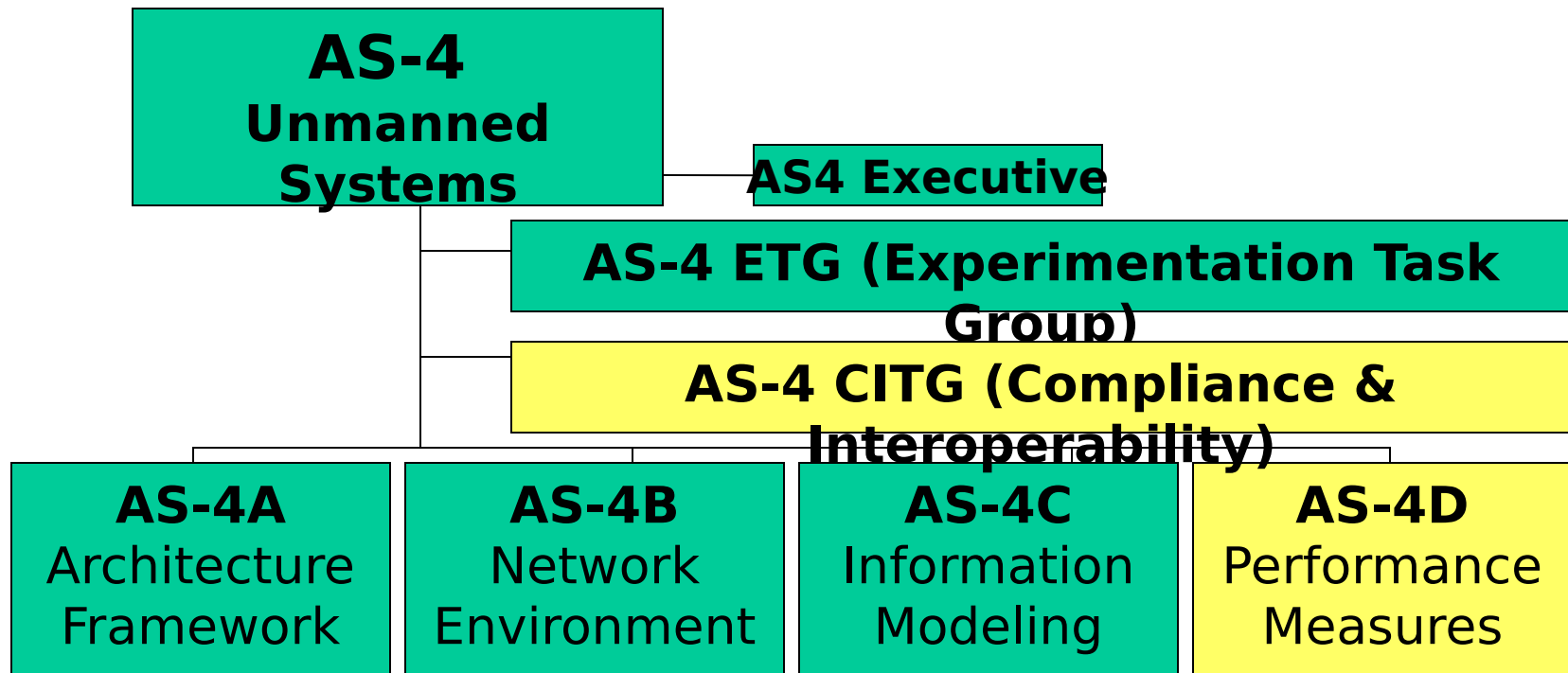
RADM William E. Landay III (then PEO-LMW, currently Chief of Naval Research) stated:

"Future UUV and USV acquisition programs will utilize JAUS for communications between the vehicle and the C&C"

Addressing potential and existing liabilities:

-Integration and development costs duplicated

-Proprietary nature of systems

-Precluded Joint-Operations and hand-off capa

12

SAE Aerospace

Avionics Systems Division (**ASD**)

AS-1 Aircraft Systems and Systems Integration	**AS-2** Embedded Computing Systems	**AS-3** Fiber Optics and Applied Photonics	**AS-4** Unmanned Systems

SAE AS-4 (JAUS)

AS-4
Unmanned Systems

AS4 Executive

AS-4 ETG (Experimentation Task Group)

AS-4 CITG (Compliance & Interoperability)

| **AS-4A** Architecture Framework | **AS-4B** Network Environment | **AS-4C** Information Modeling | **AS-4D** Performance Measures |

- Membership (October 2007)
  - 141 Members (Approx 52 Organizations)
  - 47 Voting Members; 13 Liaisons; 81 Mail List
- Balance
  User ≈ 35%; Supplier ≈ 55%; Other ≈ 10% (Academia & Labs)

14

SAE AS-4 (JAUS)

- Architecture Framework (AS-4A)
  - Common Language
  - Capabilities Supported
    - Independent of Implementation
    - Captures User Needs for Interoperabi...
- Network Environment (AS-4B)
  - Transport Interface
  - Message Packaging
    - Specified per Protocol Type
    - JAUS Message is Packet Payload
  - Optimization Considerations
- Information Modeling & Definition (A...
  - Logical Interface
  - Service Interface Specifications
    - Message Passing Rules
    - Message Content

Domains & Scope: UGV, UAV, UMV

AIR5664: JAUS History and Domain Analysis

AIR5665: Architecture Framework for Unmanned Systems (AFUS)

AIR5645: JAUS Transport Considerations Report

AS5669: JAUS Transport Specification

AS5684: JAUS Service Interface Definition Language (JSIDL)

AS5710: JAUS Service Set (JSS)

15

SAE AS-4 (JAUS)

- **Purpose**
  - **To formalize the ALFUS ad hoc working group and its results in the sustainable structure of SAE**
  - **Address terminology and metrics within AS4**
- **Charter**
  - **Terms and definitions for the performance of unmanned systems.**
  - **Measures for the performance and characterization for the unmanned systems, their components, and their interactions**
  - **Other issues related to the performance measurement of unmanned systems**
- **Origins**
  - **Started in 2003**
  - **NIST Special Publication 1011-I-1.1**
    - **Autonomy Levels for Unmanned Systems (ALFUS) Framework**
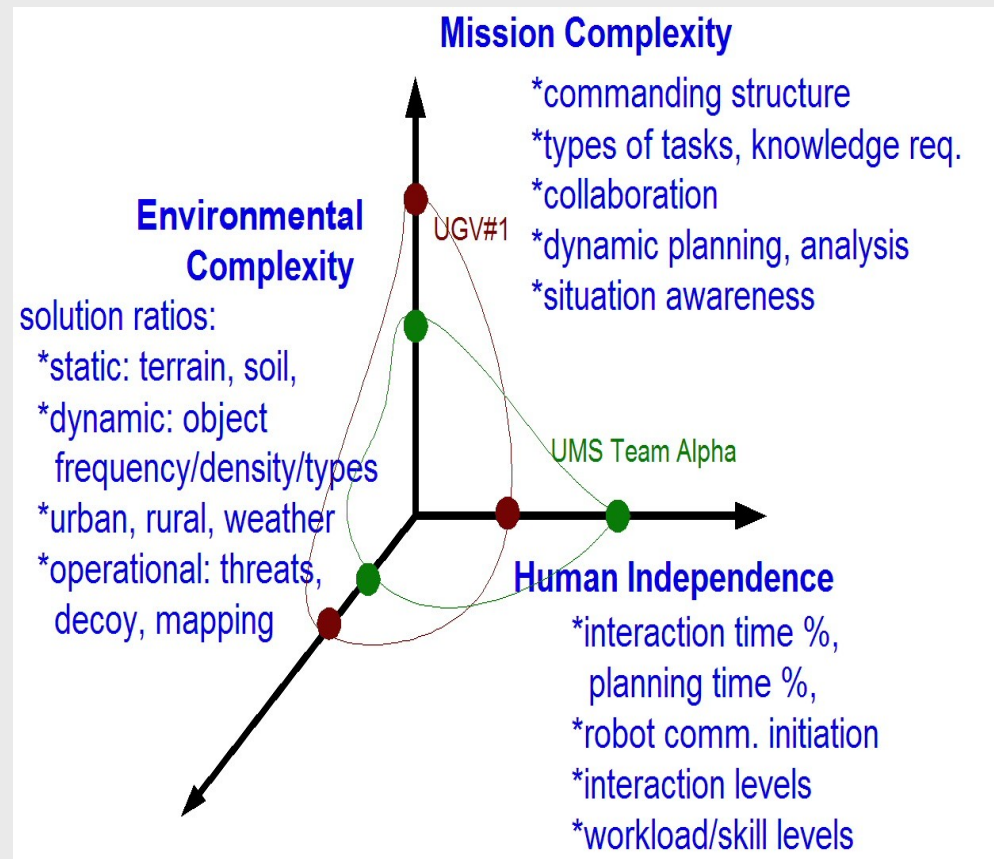   - **Volume I: Terminology, Version 1.1**

Autonomy

A UMS's own ability of integrated sensing, perceiving, analyzing, communicating, planning, decision-making, and acting/executing, to achieve its goals as assigned.

Contextual Autonomous Capability (CAC)

A UMS's CAC is characterized by the missions that the system is capable of performing, the environments within which the missions are performed, and human independence that can be allowed in the performance of the missions.
Each of the aspects, or axes, namely, mission complexity (MC), environmental complexity (EC), and human independence (HI) is further attributed with a set of metrics to facilitate the specification, analysis, evaluation, and measurement of the CAC of particular UMSs.
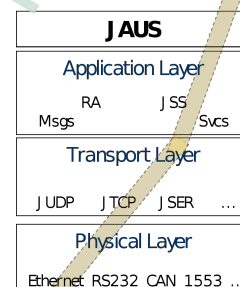
Mission Complexity
*commanding structure
*types of tasks, knowledge req.
*collaboration
*dynamic planning, analysis
*situation awareness

Environmental Complexity
UGV#1
solution ratios:
*static: terrain, soil,
*dynamic: object
   frequency/density/types
*urban, rural, weather
*operational: threats,
   decoy, mapping

UMS Team Alpha

Human Independence
*interaction time %,
   planning time %,
*robot comm. initiation
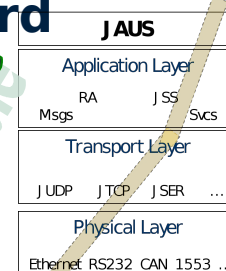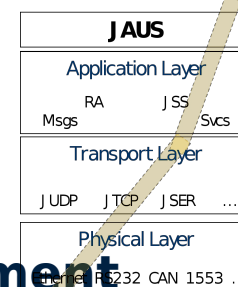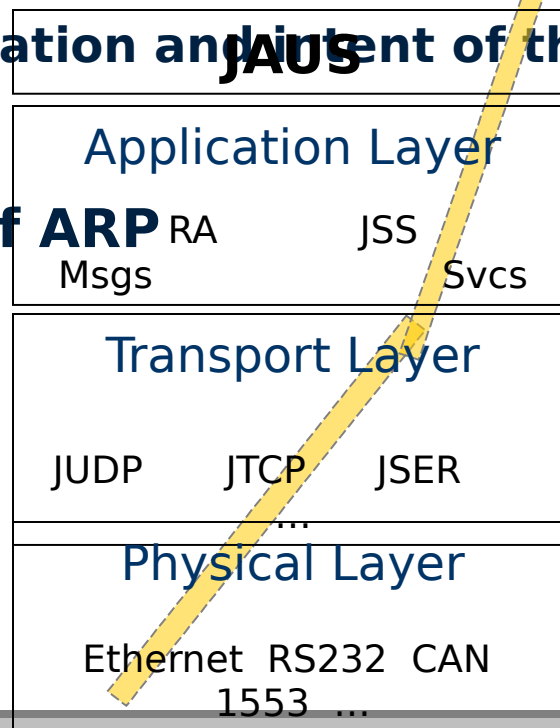*interaction levels
*workload/skill levels

17

Compliance & Interoperability Policy Task Group

- **Purpose**
  - **JAUS is a collection of standards**
  - **Provide guidance for writing a JAUS requirement**
  - **Clarify both the application and intent of the standard**
- **Product**
  - **Near term release of ARP**
  - **JAUS *al la Carte'***

JAUS

Application Layer

RA   JSS
Msgs   Svcs

Transport Layer

JUDP   JTCP   JSER   …

Physical Layer

Ethernet RS232 CAN 1553 …

JAUS

Application Layer

RA   JSS
Msgs   Svcs

Transport Layer

JUDP   JTCP   JSER   …

Physical Layer

Ethernet RS232 CAN 1553 …

JAUS

Application Layer

RA   JSS
Msgs   Svcs

Transport Layer

JUDP   JTCP   JSER   …

Physical Layer

Ethernet RS232 CAN 1553 …

JAUS

Application Layer

RA   JSS
Msgs   Svcs

Transport Layer

JUDP   JTCP   JSER

…

Physical Layer

Ethernet RS232 CAN 1553 …

Interoperable

For more information please see

www.jauswg.org

- Current JAUS documents
- Announcements for future meetings

**Ralph "Woody" English
DeVivo AST, Inc.
woodyenglish@devivoast.com**

JAUS

19

SAE AS-4 (JAUS)

- **Introduction**
- **Scope**
- **Background**
- **Organization**
- **Document Set & Standards**
- **Future Directions**
  → **Backup Slides**

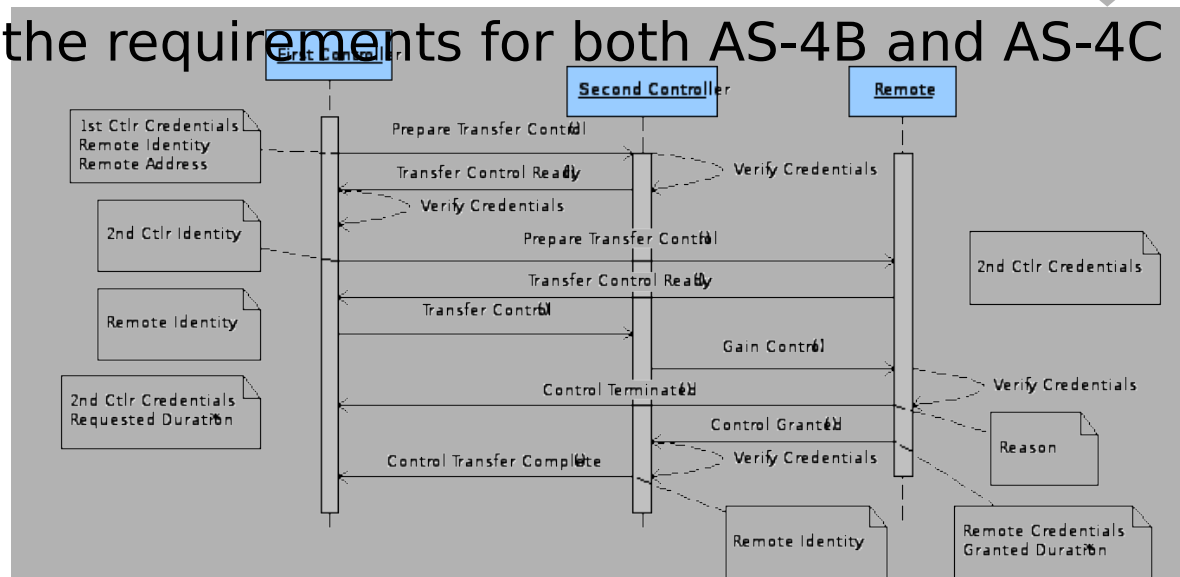- AFUS defines the unmanned system capabilities and relationships
  - Discovery
  - Authority and access
  - Mobility
  - Planning
  - Weapons
  - World models
  - UUV unique
  - ...

Examples

- Serves as the requirements for both AS-4B and AS-4C

SAE AS-4 (JAUS)

JAUS RA User Feedback

- **<u>Specificity</u>** – reduce the potential of misinterpretation during implementation
- **<u>Automation</u>** – automate various aspects of the development of a JAUS system
- **<u>Completeness</u>** – incorporate all essential elements necessary for application level interoperability
- **<u>Correctness</u>** – thoroughly verify the essential elements specified

JAUS Service
Interface Definition Language

| JAUS Service Set | ≈ | Reference Architecture (message | ➕ | Machine-Readable | ➕ | Formal Methods |

Functional Equivalence

Automation and Specificity

Complete and Correct

22
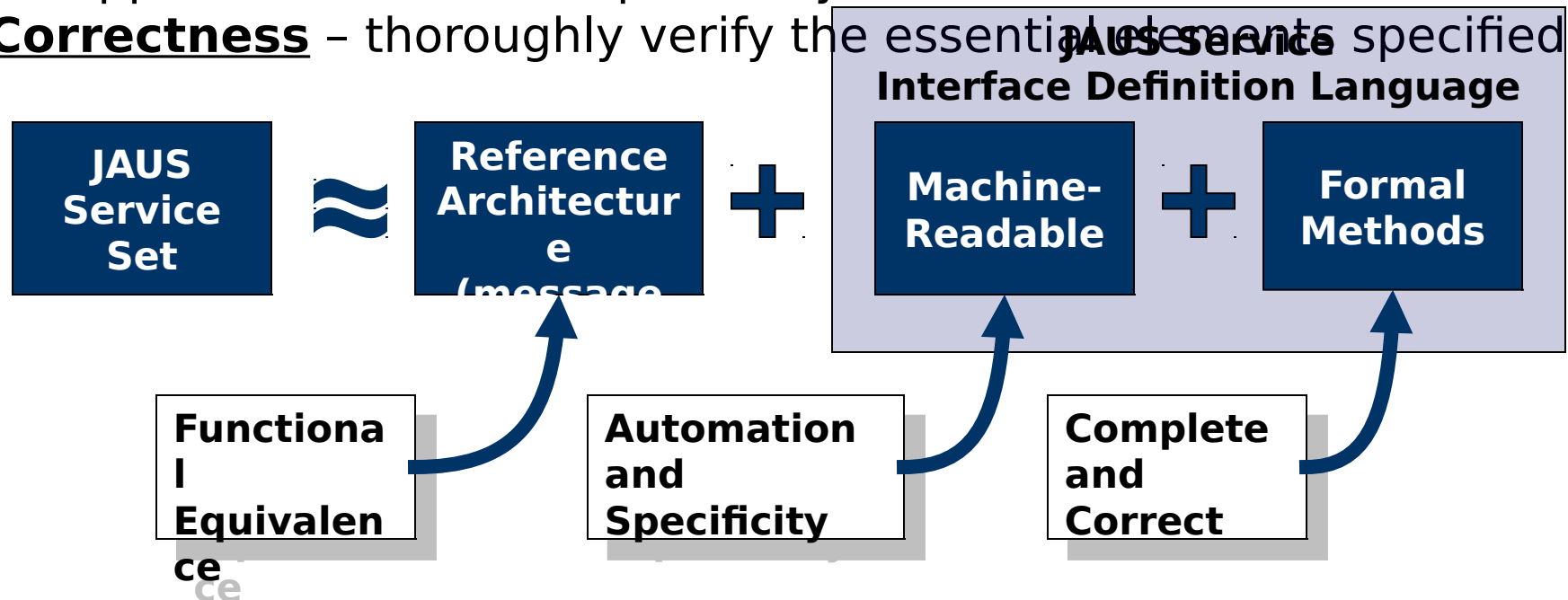
Use Machine Readable Language...

- Precise semantics eliminates misinterpretation

- Precise syntax and semantics allows for the development of automated tools

- Forces completeness in specification by clearly stating what information is required and what is optional

- AS5684 (JSIDL)

  – Defines a language for specifying service interfaces

  – Uses a schema language for XML

    • Defines a formal structure to create messages

    • Uses finite state machines to define procedure rules

  – Key features:

    • Allows service interface definitions from the perspective of a server, a client, or a server and its client(s)

    • Separates application behavior from the interface behavior

    • Encourages reuse of existing services and data types

    • Does not force a particular implementation

Define Essential Elements in a Protocol...*

Service to be provided by the protocol

Assumptions about the environment in which the protocol is executed

Vocabulary of messages used to implement the protocol

Encoding (format) of each message in the vocabulary

Procedure Rules guarding the consistency of message exchanges

G.J. Holzmann, Design and Validation of Computer Protocols, Prentice Hall Software Series, 1991

AS5710 Service Design Principles

• **Loosely Coupled** – minimize dependencies between services

• **Abstract** – hide logic other than what is defined in the interface

• **Autonomous** – each service has control over the logic they encapsulate

• **Reusable** – divide logic into services such that it promotes reuse

• **Extensible** – allow for future growth

24

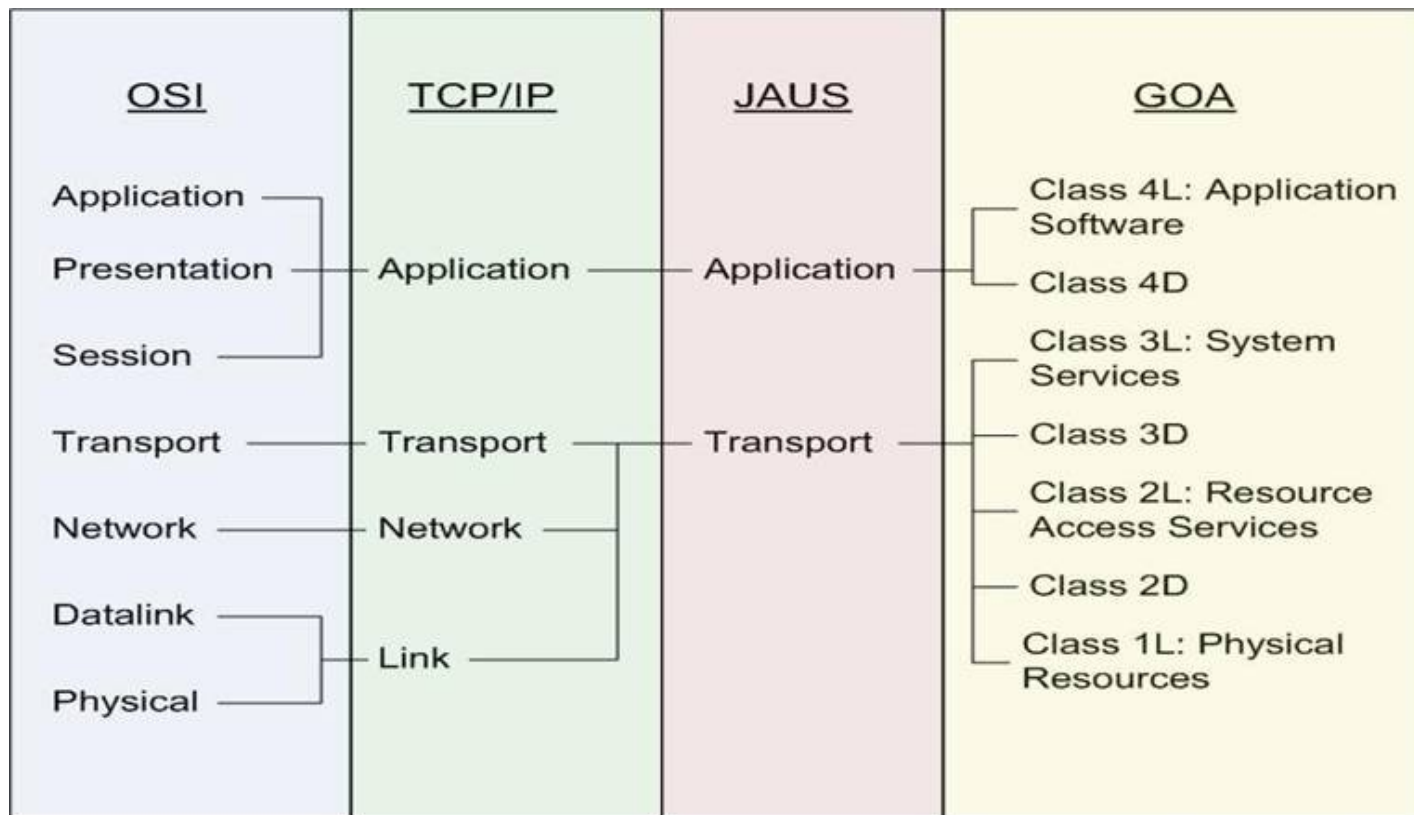## JAUS System Topology and Service Interface

System

Subsystem **Subsystem** ... **Subsystem**

Node **Node** **Node** ... **Node**

Component **Component** ... **Component**

Service	Service	Service
Service	Service	Service
⋮	⋮	⋮
Service	Service	Service

"A subsystem is composed of component software, distributed across one or more nodes.

Service interfaces are specified with a machine readable language using formal methods

Components are comprised of one or

Service ⟵ **Description Assumptions Vocabulary Encoding Procedure Rules** ⟶ **Service**

25

OSI	TCP/IP	JAUS	GOA
Application			Class 4L: Application Software
Presentation	Application	Application	Class 4D
Session			Class 3L: System Services
Transport	Transport	Transport	Class 3D
Network	Network		Class 2L: Resource Access Services
Datalink			Class 2D
Physical	Link		Class 1L: Physical Resources

OSI = Open Systems Interconnect;

GOA = Generic Open Architecture;

TCP/IP = Transmission Control Protocol / Internet Protocol

JAUS on the Transport Protocol Stack

JAUS defines a simple stack, as befits a standard focused on application semantics, not transport details – again, focus is on the interoperable and efficient implementation of application semantics